

BROWSER-CONTROLLED SCANNING SYSTEM AND METHOD

FIELD OF THE INVENTION

5 The present disclosure relates to a browser-controlled scanning system and method. More particularly, the disclosure relates to a system and method in which the operation of a scanning device is controlled such that documents can be scanned and, where desired, optically character recognized, and then displayed to the user with a browser.

BACKGROUND OF THE INVENTION

10 Peripheral devices are adapted to be accessed and used by computing devices such as a personal computer (PC). Traditionally, printers were accessible in this manner while other “office” devices were only configured for “walk-up” use. With the recent
15 focus on networking technology, however, many other devices can be accessed and used with a host computing device. For example, photocopiers, facsimile machines, scanners, multifunction peripherals (MFPs) capable of several different functionalities traditionally conducted by separate devices, network appliances, *etc.* are currently available that are configured for this type of use.

20 To enable such access and control, one or more software applications normally must be stored on the user’s computing device. Such applications typically comprise a user interface and one or more device drivers. The user interface is provided as a means

for receiving user commands and selections regarding the tasks the user wishes to be completed by the end device and the device drivers are configured to send jobs from the computing device to the end device to fulfill the requested tasks.

Typically, each end device to be accessed by the computing device has its own
5 separate software application. Moreover, separate software applications are typically
needed for each different functionality the end device performs where it performs more
than one functionality (*e.g.*, scanning, faxing, copying, and printing). Normally, these
software applications are not standardized. Therefore, the layout of the user interface
and the manner in which the end device is controlled may be different for each
10 application, even for different devices made by the same manufacturer and for single
devices that provide multiple functionalities.

The arrangement described above presents several disadvantages to the user as
well as the device manufacturer. With regard to the user, the user must install separate
software for each different device and/or functionality the user plans to use. In addition,
15 the user may need to update this software when new software becomes available from
the device manufacturer (*e.g.*, updated driver software). This is very time-consuming
for the user and places a burden upon the user to keep apprised of any software
improvements that have been made by the device manufacturer. Once the software has
been installed by the user, the user must become familiar with each different software
20 application, both in terms of the user interface and the manner in which the software is
used to control the device and/or functionality. This can be frustrating for the user,
particularly where the user must access many different devices and/or functionalities.

In terms of the device manufacturer, disadvantages include having to reconfigure
the software as the underlying operating environment (*e.g.*, WindowsTM, UnixTM) is

changed by third parties as well as having to provide new software (for any purpose) to the various purchasers of a given device as the new software is developed. Furthermore, device manufacturers normally must provide customer support for all versions of software that have been produced in that some users may still have old
 5 versions of the software. In some situations, valuable time may be lost in just determining what software the user possesses.

From the foregoing, it can be appreciated that it would be desirable to have a system and method for accessing and using a device that avoids one or more of the difficulties identified above.

10

SUMMARY OF THE INVENTION

The present disclosure relates to a browser-controlled scanning system and method. In one arrangement, the system is adapted for and method comprises receiving a scan request from a user browser, uploading content to the user browser,
 15 receiving selections made with the user browser, and scanning the document in accordance with the user selections.

The disclosure further relates to a scanning device. In one embodiment, the scanning device comprises a processing device, scanning hardware, and memory comprising a scan control module and an embedded server, the scan control module
 20 comprising a scanning module and an optical character recognition module, the scan control module further including logic for generating at least one control screen that can be uploaded to a user browser.

Other systems, methods, features, and advantages of the invention will become apparent upon reading the following specification, when taken in conjunction with the accompanying drawings.

5

BRIEF DESCRIPTION OF THE DRAWINGS

The invention can be better understood with reference to the following drawings. The components in the drawings are not necessarily to scale, emphasis instead being placed upon clearly illustrating the principles of the present invention.

FIG. 1 is a schematic view of an example browser-controlled scanning system.

10

FIG. 2 is a schematic view of a computing device shown in FIG. 2.

FIG. 3 is a schematic view of a scanning device shown in FIG. 2.

FIG. 4 is a flow diagram that illustrates use of the browser of the computing device shown in FIG. 3 in controlling the scanning device.

15

FIG. 5 is a schematic representation of content that is uploaded into the browser shown in FIG. 2.

FIGS. 6 is an example control screen that can be presented to the user with the browser shown in FIG. 2.

FIG. 7 is a flow diagram that illustrates operation of a scan control module of the scanning device shown in FIG. 3.

20

DETAILED DESCRIPTION

Disclosed herein is a scanning system and method that is controlled with the user's browser. With this system and method, a browser is used to receive content that, as is described in greater detail below, can comprise one or more pages or screens that

can be used to control a scanning device. In addition, this content may further comprise small applications (*e.g.*, Java applets) that are embedded in the hypertext markup language (HTML) code of the page displayed by the browser that are configured to perform various designated tasks for the scanning device.

5 The browser can be used to initiate scanning and, where desired, performance of optical character recognition (OCR) on the various image data scanned by the scanning device. Where only scanning is desired, the scanned document can then be viewed as an image having one of various different formats with the browser. Where OCR is desired, OCR can be performed and a resulting HTML document can be viewed with the
10 browser. Operating in this manner, the user can control all scanning, OCR, and file conversion with the user's browser. As can be appreciated from this brief explanation, this arrangement does not require installation of a separate application and streamlines the scanning/OCR processes for the user while simultaneously utilizing a user interface with which the user is already familiar.

15 To facilitate description of the invention, an example browser-controlled scanning system will first be discussed with reference to the figures. Although this system is described in detail, it will be appreciated that this system is provided for purposes of illustration only and that various modifications are feasible without departing from the inventive concept. After the example system has been described,
20 examples of operation of the system will be provided to explain the manners in which scanning/OCR control can be achieved.

Referring now in more detail to FIG. 1, illustrated is an example browser-controlled scanning system 100. As indicated in this figure, the system 100 generally comprises a computing device 102 and one or more scanning devices 104. As shown in

FIG. 1, the computing device 102 can comprise a personal computer (PC). However, it is to be understood that the computing device 102 can comprise substantially any device that can be used to access and use a scanning device. Therefore, the computing device could, alternatively, comprise a laptop computer, personal digital assistant (PDA), mobile telephone, *etc.* For the purposes of this disclosure, the term “scanning device” is used to denote any device that is capable of electronically scanning data. Therefore, the scanning device 104 can, for instance, comprise an independent scanner 106 or a multifunction peripheral (MFP) 108, sometimes referred to as an “all-in-one,” that is capable of scanning as well as other different functionalities.

As is further identified in FIG. 1, the computing device 102 and the scanning devices 104 can be connected to a network 110. The network 110 typically comprises one or more sub-networks that are communicatively coupled to each other. By way of example, these networks can include one or more local area networks (LANs) and/or wide area networks (WANs). Indeed, in some embodiments, the network 110 may comprise a set of networks that forms part of the Internet. As is also depicted in FIG. 1, the computing device 102 can, optionally, be directly connected to one or both of the scanning devices 104. Such an arrangement is likely in a home or small office environment in which the user does not have access to a network and instead directly communicates to a scanning device 104. In such a scenario, communication can be facilitated with a direct electrical and/or optical connection or through wireless communication.

FIG. 2 is a schematic view illustrating an example architecture for the computing device 102 shown in FIG. 1. As indicated in FIG. 2, the computing device 102 can comprise a processing device 200, memory 202, one or more user interface

devices 204, a display 206, one or more I/O devices 208, and one or more networking devices 210, each of which are connected to a local interface 212. The processing device 200 can include any custom made or commercially available processor, a central processing unit (CPU) or an auxiliary processor among several processors associated with the computing device 102, a semiconductor based microprocessor (in the form of a microchip), or a macroprocessor. The memory 202 can include any one of a combination of volatile memory elements (*e.g.*; random access memory (RAM, such as DRAM, SRAM, *etc.*)) and nonvolatile memory elements (*e.g.*, ROM, hard drive, tape, CDROM, *etc.*).

The one or more user interface devices 204 comprise those components with which the user can interact with the computing device 102. Where the computing device 102 comprises a PC or similar device, these components can comprise those typically used in conjunction with a PC such as a keyboard and mouse. Where the computing device 102 comprises a handheld device such as a PDA or mobile telephone, the user interface devices 204 can comprise one or more function buttons or keys. The display 206 can comprise a display typically used in conjunction with a PC such as a computer monitor or plasma screen. Where the computing device 102 comprises a handheld device, the display 206 can comprise a liquid crystal display (LCD) that may or may not be touch-sensitive.

The one or more I/O devices 208 comprise components used to facilitate connection of the computing device 102 to other devices directly, such as the scanning devices 104. Therefore, these devices can, for instance, comprise one or more serial, parallel, small system interface (SCSI), universal serial bus (USB), IEEE 1394 (*e.g.*, FirewireTM), or personal area network (PAN) connection devices. The networking

devices 210 comprise the various components used to transmit and/or receive data over the network 110. By way of example, the networking devices 210 include a device that can communicate both inputs and outputs, for instance, a modulator/demodulator (*e.g.*, modem), a radio frequency (RF) or other transceiver, a telephonic interface, a bridge, a router, as well as a network card, *etc.*

The memory 202 normally comprises various software programs including an operating system 214 and a user browser 216. Although various other software programs may be stored in memory 202, they are typically not required to obtain the scanning and OCR control that is the subject of the present disclosure and therefore have not been identified. The operating system 214 controls the execution of other software, such as the browser 216, and provides scheduling, input-output control, file and data management, memory management, and communication control and related services. The browser 216 comprises the software that is used to browse data over the network 110 and, as described in greater detail below, thereby access and use the scanning devices 104. The browser 216 typically comprises various different components such as a user application that the user can run on the computing device 102 to interface with the browser software. The browser can, for example, comprise a currently available Internet browser such as Microsoft Internet ExplorerTM or Netscape NavigatorTM.

FIG. 3 is a schematic view illustrating an example architecture for the scanning devices 104 shown in FIG. 1. As indicated in FIG. 3, each scanning device 104 can comprise a processing device 300, memory 302, scanning hardware 304, one or more user interface devices 306, one or more I/O devices 308, and one or more networking devices 310. Each of these components is connected to a local interface

312 that, by way of example, comprises one or more internal buses. The processing device 300 is adapted to execute commands stored in memory 302 and can comprise a general-purpose processor, a microprocessor, one or more application-specific integrated circuits (ASICs), a plurality of suitably configured digital logic gates, and other well known electrical configurations comprised of discrete elements both individually and in various combinations to coordinate the overall operation of the scanning device 104.

The scanning hardware 304 comprises the components with which the scanning device 104 can create an electronic copy of a hardcopy document.

10 Accordingly, the scanning hardware 304 can comprise, for instance, a paper drive mechanism, light source (*e.g.*, fluorescent light), light-sensing devices (*e.g.*, charge-coupled devices (CCDs)), and various optics (*e.g.*, lenses, mirrors). The one or more user interface devices 306 typically comprise interface tools with which the device settings can be changed and through which the user can communicate commands directly to the scanning device 104. By way of example, the user interface devices 306 comprise one or more function keys and/or buttons with which the operation of the scanning device 104 can be controlled, and a display, such as a liquid crystal display (LCD), with which information can be visually communicated to the user. Finally, the I/O devices 308 and networking devices 310 can have configurations similar to like-named components identified above with reference to FIG. 2.

The memory 302 includes various software (*e.g.*, firmware) programs including an operating system 314, scan control module 316, and an embedded server 318. The operating system 314 contains the various commands used to control the general operation of the scanning device 104. The scan control module 316 comprises the

various code used to control the operation of the scanning hardware 304 in response to commands received from the user (*e.g.*, over the network 110). As indicated in FIG 3, the scan control module 316 can include both a scanning module 320 that is configured to, in conjunction with the scanning hardware 304, scan documents into electronic form, and an OCR module 322 that is configured to recognize characters of the scanned document. The operation of the scan control module 316 is described in detail with respect to FIG. 7 below. The server 318 comprises the software (*e.g.*, firmware) that is used to serve-up data to browsers that request the data. By way of example, the data can comprise one or more pages or control screens and one or more small programs that are configured to perform designated tasks.

Various software and/or firmware programs have been described herein. It is to be understood that these programs can be stored on any computer readable medium for use by or in connection with any computer related system or method. In the context of this document, a computer readable medium is an electronic, magnetic, optical, or other physical device or means that can contain or store a computer program for use by or in connection with a computer related system or method. These programs can be embodied in any computer-readable medium for use by or in connection with an instruction execution system, apparatus, or device, such as a computer-based system, processor-containing system, or other system that can fetch the instructions from the instruction execution system, apparatus, or device and execute the instructions. In the context of this document, a “computer-readable medium” can be any means that can store, communicate, propagate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device.

The computer readable medium can be, for example but not limited to, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, device, or propagation medium. More specific examples (a nonexhaustive list) of the computer-readable medium include an electrical connection having one or
5 more wires, a portable computer diskette, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM, EEPROM, or Flash memory), an optical fiber, and a portable compact disc read-only memory (CDROM). Note that the computer-readable medium can even be paper or another suitable medium upon which a program is printed, as the program can be
10 electronically captured, via for instance optical scanning of the paper or other medium, then compiled, interpreted or otherwise processed in a suitable manner if necessary, and then stored in a computer memory.

An example system 100 having been described above, operation of the system will now be discussed. In the discussion that follows, flow diagrams are provided. It
15 is to be understood that any process steps or blocks in these flow diagrams represent modules, segments, or portions of code that include one or more executable instructions for implementing specific logical functions or steps in the process. It will be appreciated that, although particular example process steps are described, alternative implementations are feasible. Moreover, steps may be executed out of
20 order from that shown or discussed, including substantially concurrently or in reverse order, depending on the functionality involved.

As noted above, the system 100 generally operates so as to facilitate control of a scanning device with the user's browser, for example, browser 216. An example of the operation of the browser 216 as used to control a scanning device 104 is provided in

FIG. 4. Beginning with block 400 of this figure, the browser 216 is first activated. This activation can occur in a variety of different ways. Typically, however, activation occurs in response to the user opening the browser 216 from the system desktop. In any case, once activated, a scan request can be received from the user, as indicated in block 402.

5 This request can be transmitted via the network 110 or through a direct connection. In that scanning will ultimately be conducted by the scanning device 104, the scan request is also a request to access the scanning device. The scan request can be entered in a variety of ways. In a simplified case, the user can have entered the network address of the server 318 of the scanning device 104. By way of example, this address can
10 comprise a universal resource locator (URL) that the browser 216 can use to make calls to the scanning device server 318.

In another case, the user can have selected a “scanning device” link stored within a “favorites” listing that forms part of the browser 216. By way of example, this link could have been created manually by the user, automatically added to the
15 favorites list by content uploaded to the browser 216 when the server 318 is first accessed with the browser, *etc.* In a further case, the user can have selected a “scan” button provided on the tool bar of the browser 216 which again was either manually added by the user or automatically added by content uploaded to the browser.

Once the scan request has been received, and the address of the scanning
20 device 104 therefore has been provided, the browser 216 transmits a connection request to the server 318 of the scanning device 104, as indicated in block 504. This request is then received by the server 318 and the server maps the address of the browser 216 to one or more pages. Through this action, various content is uploaded to the browser 216 and is therefore “received” by the browser, as indicated in block

406. As will be appreciated by persons having ordinary skill in the art, the nature of this content depends upon the operations desired.

A schematic representation of the uploaded content is provided in FIG. 5. As indicated in this figure, the content 500 can comprise a user interface 502, such as a graphical user interface (GUI), with which the user can make selections to communicate commands to the scanning device 104. This user interface 502 is configured to present to a series of pages or control screens to the user that are viewable in a viewing window of the browser 216. In addition to the interface 502, the content 500 can comprise a plurality of small applications 504, generally referred to as applets (*e.g.*, Java applets), that are configured to perform various tasks. For example, as discussed below, one application 504 can be configured to perform to OCR while another can be configured to search for particular language OCR modules on the computing device 102.

Once the content 500 has been received, the user interface is presented to the user, as indicated in block 408. Where the interface comprises a GUI, one or a series of pages or control screens can be displayed to the user in the viewing window of the browser 216. FIG. 6 provides an example control screen that can be displayed to the user. In particular, FIG. 6 illustrates a scan menu screen 600 that can be displayed to the user. By way of example, this menu screen 600 can be the first screen that is presented to the user. However, it is to be understood that this menu screen 600 need not necessarily be the first. For instance, where the scanning device 104 is capable of performing other functions (*e.g.*, printing), the first screen presented to the user may request the user to designate which of the particular available device functionalities is to be accessed.

As is apparent from FIG. 6, the scan menu screen 600 can present the user with several selectable options 602. These options can include, for example, “Standard Scan” and “OCR Scan.” As their names suggest, the “Standard Scan” option pertains to scanning a document only while “OCR Scan” pertains to scanning a document and then conducting OCR on the document. As indicated in FIG. 6, the scan menu screen 600 can further include check “boxes” 604 that the user can mark to convey the user’s selection. Once the user is satisfied with his or her selection, the user can select a “Continue” button 606 that is provided on the scan menu screen 600.

Returning now to FIG. 4, the browser 216 can receive the user’s selection, as indicated in block 410, and transmit the selection to the scanning device 104, as indicated in block 412. Although not shown in the figures, it will be appreciated that other pages or control screens could first be presented to the user, if desired. For example, after completing the scan menu screen 600, the user can be provided with an options screen with which the user can specify various other scanning related information (*e.g.*, scan resolution, contrast, scaling, down sampling, *etc.*).

At this point, the scanning selection(s) can be received by the scanning device 104 and the requested scanning, and OCR if desired, can be performed. Referring now to FIG. 7, illustrated is an example of operation of the scanning device 104 in this capacity. More particularly, FIG. 7 illustrates an example of the operation of the scan control module 316 and the server 318 of the scanning device 104, which work in concert to perform the desired functionalities. The server 318, and therefore scan control module 316, first receives the connection request from the browser 216 in the manner described above, as indicated in block 700. At this point, the scan control module 316 identifies the content that is to be provided to the browser 216 so that, as

5

10

20

typical scenario, the OCR can be directly performed by the OCR module 322 of the scan control module 316. Alternatively, however, the OCR can be performed by an application 504 that has been uploaded to the browser. In yet another alternative, OCR can be performed by an OCR module (not shown) of the computing device 102

5 where the device comprises such a module. Such a scenario may occur where, for instance, the OCR module 322 of the scanning device is configured for recognizing English text only and the user desires recognition of text in a different language. This desire could, for example, be identified explicitly by the user as a scan selection on one of the control screens, or can be inferred by the scan control module 316 where

10 the computing device operating system 214 is configured for use in a different spoken language. In the latter case, the scan control module 316 can cause an application 504 (*e.g.*, Java applet) to be uploaded to the browser 218 that is configured to search for such an appropriate OCR module on the computing device. Once located, the OCR module can be called upon by the application 504 to perform the needed OCR.

15 After OCR has been performed, the document can be stored as a file having a text format that is readily supported by the browser 216 (*e.g.*, HTML). Where the scanning and OCR were performed by the scanning device 102, the OCR document file can be stored on an internal hard drive of the scanning device. Alternatively, where the OCR was performed by an OCR module of the computing device 102, the

20 document file can be stored in memory 202 (*e.g.*, RAM) until the user determines what the user would like to do with the document.

Referring now to block 714, scan control module 316 can facilitate display of the scanned document file or scanned and OCR'ed document file with the browser 216. Where the document was only scanned, the user can view the document in the

browser viewing window as an image. For example, each page of the scanned document can be presented to the user in thumbnail form and the full page version displayed when each individual thumbnail is selected. Alternatively, each page can be presented one-by-one with each further page being accessed by the user by selecting a
5 “next” button. In a further alternative, each scanned page of can be provided on one “page” such that each document page is viewed by scrolling downwardly from the first document page to the last.

Where the document was scanned and OCR was performed, the user can access the text of the document. If the user would like to modify the text, he or she
10 can save the HTML document file in the conventional manner and later open the document in an appropriate word processing application (*e.g.*, Microsoft WordTM). Once opened in this manner, the document can be stored to the computing device hard drive as a word processing file and treated accordingly.

As will be appreciated from the above discussion, several advantages are
15 provided with the disclosed system and method. First, in that all the software necessary for providing scan control is stored on the scanning device 104 and uploaded from that device to the user’s browser 216, there are no software applications for the user to download. In that the user’s browser 216 is used as the user interface, the user further does not have to become accustomed to disparate user
20 interfaces of many different applications, thereby providing interface standardization. Furthermore, due to the centralization of the software, any software updates can be implemented on the scanning device 104 alone but will be available to all users immediately. Moreover, in that the user’s browser is used independently of the user’s operating environment, the scanning device manufacturer need not rewrite the

software every time a third party software manufacturer updates its operating system.

Although these advantages have been identified, persons having ordinary skill in the art will appreciate that other advantages exist. Furthermore, such persons will appreciate that, depending upon the particular embodiment that is implemented, one

5 or more of these advantages may not necessarily apply.

While particular embodiments of the invention have been disclosed in detail in the foregoing description and drawings for purposes of example, it will be understood by those skilled in the art that variations and modifications thereof can be made without departing from the scope of the invention as set forth in the following claims.